

Державний вищий навчальний заклад
Донецький національний технічний університет
Кафедра прикладної математики та інформатики

«ЗАТВЕРДЖУЮ»

Перший проректор

_____ Леонід Бачурін

« _____ » _____ 2023 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ДВС 1.03 СУЧАСНІ ТЕХНОЛОГІЇ ПРОГРАМУВАННЯ

(шифр і назва навчальної дисципліни)

Рівень освіти:	перший (бакалаврський)
Спеціальність	<u>122 Комп'ютерні науки</u> (шифр і назва спеціальності)
Освітня програма	<u>Комп'ютерні науки</u> (назва освітньої програми)
Мова навчання:	українська

Луцьк – 2023

Робоча програма з дисципліни ”Сучасні технології програмування “
для здобувачів вищої освіти за спеціальністю 122 – Комп’ютерні науки

« 30 » серпня 2023 року. – __ с.

Розробник:

Башков Є.О., д.т.н., професор кафедри ПМІ, професор



Робоча програма затверджена на засіданні кафедри **прикладної математики та інформатики**
Протокол № 8 від “ 31 ” серпня 2023 р.

В.о. завідувача кафедри **прикладної математики та інформатики**



(підпис)

(Маслова Н.О.)

(прізвище та ініціали)

Схвалено науково-методичною комісією з галузі знань 12 « Інформаційні технології »
Протокол № 5 від. “ 01 ” вересня 2023 р.

Голова



(підпис)

(Башков Є.О.)

(прізвище та ініціали)

Загальна інформація

Форма навчання	Денна
Статус	Вибіркова
Обсяг в кредитах ЄКТС	6
Обсяг в годинах за навчальним планом, разом: в тому числі:	180
лекції:	32
практичні заняття:	32
лабораторні заняття:	-
семінари:	-
самостійна робота: Індивідуальне завдання	116 Розрахункова робота
Форма підсумкового контролю	Іспит
Дисципліну викладають	Викладач – Башков Євген Олександрович, https://donntu.edu.ua/knt/pmi , yevhen.bashkov@donntu.edu.ua

Передумови для вивчення дисципліни: успішному вивченню дисципліни «Сучасні технології програмування» сприяє попереднє опанування такими дисциплінами, як «Основи алгоритмізації», "Програмування", "Об'єктно – орієнтовне програмування".

2. Мета вивчення навчальної дисципліни «Сучасні технології програмування»

Дисципліна «Сучасні технології програмування» має на меті отримання майбутнім фахівцем знань та навиків, які необхідні для проектування та розробки програмних додатків на основі сучасних розвинутих мов та технологій програмування.

Компетентності:

ЗК 01. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК 02. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК 03. Здатність працювати в міжнародному контексті.

ЗК 04. Здатність розв'язувати комплексні проблеми комп'ютерних наук на основі системного наукового світогляду та загального культурного кругозору із дотриманням принципів професійної етики та академічної доброчесності.

ФК 01. Здатність виконувати оригінальні дослідження, досягати наукових результатів, які створюють нові знання у комп'ютерних науках та дотичних до них міждисциплінарних напрямках і можуть бути опубліковані у провідних наукових виданнях з комп'ютерних наук та суміжних галузей.

ФК 02. Здатність застосовувати сучасні методології, методи та інструменти експериментальних і теоретичних досліджень у сфері комп'ютерних наук, сучасні цифрові технології, бази даних та інші електронні ресурси у науковій та освітній діяльності.

ФК 03. Здатність виявляти, ставити та вирішувати дослідницькі науково-прикладні задачі та/або проблеми в сфері комп'ютерних наук, оцінювати та забезпечувати якість виконуваних досліджень.

ФК 04. Здатність ініціювати, розробляти і реалізовувати комплексні інноваційні проекти у галузі комп'ютерних наук та дотичні до неї міждисциплінарних проєктах, демонструвати лідерство під час їх реалізації.

ФК 05. Здатність здійснювати науково-педагогічну діяльність у вищій освіті у сфері комп'ютерних наук.

ФК 06. Здатність аналізувати та оцінювати сучасний стан і тенденції розвитку комп'ютерних наук та інформаційних технологій.

Програмні результати навчання:

ПРН 01. Мати передові концептуальні та методологічні знання з комп'ютерних наук і на межі предметних галузей, а також дослідницькі навички, достатні для проведення наукових і прикладних досліджень на рівні останніх світових досягнень з відповідного напрямку, отримання нових знань та/або здійснення інновацій.

ПРН 02. Вільно презентувати та обговорювати з фахівцями і нефахівцями результати досліджень, наукові та прикладні проблеми комп'ютерних наук державною та іноземною мовами, оприлюднювати результати досліджень у наукових публікаціях у провідних міжнародних наукових виданнях.

ПРН 03. Формулювати і перевіряти гіпотези; використовувати для обґрунтування висновків належні докази, зокрема, результати теоретичного аналізу, експериментальних досліджень і математичного та/або комп'ютерного моделювання, наявні літературні дані.

ПРН 04. Розробляти та досліджувати концептуальні, математичні і комп'ютерні моделі процесів і систем, ефективно використовувати їх для отримання нових знань та/або створення інноваційних продуктів у комп'ютерних науках та дотичних міждисциплінарних напрямках.

ПРН 05. Планувати і виконувати експериментальні та/або теоретичні дослідження з комп'ютерних наук та дотичних міждисциплінарних напрямів з використанням сучасних інструментів, критично аналізувати результати власних досліджень і результати інших дослідників у контексті усього комплексу сучасних знань щодо досліджуваної проблеми.

ПРН 06. Застосовувати сучасні інструменти і технології пошуку, оброблення та аналізу інформації, зокрема, статистичні методи аналізу даних великого обсягу та/або складної структури, спеціалізовані бази даних та інформаційні системи.

ПРН 07. Розробляти та реалізовувати наукові та/або інноваційні інженерні проекти, які дають можливість переосмислити наявне та створити нове цілісне знання та/або професійну практику і розв'язувати значущі наукові та технологічні проблеми комп'ютерної науки з дотриманням норм академічної етики і врахуванням соціальних, економічних, екологічних та правових аспектів.

ПРН 08. Визначати актуальні наукові та практичні проблеми у сфері комп'ютерних наук, глибоко розуміти загальні принципи та методи комп'ютерних наук, а також методологію наукових досліджень, застосовувати їх у власних дослідженнях у сфері комп'ютерних наук та у викладацькій практиці.

ПРН 09. Вивчати, узагальнювати та впроваджувати в навчальний процес інновації комп'ютерних наук.

ПРН 10. Відшукувати, оцінювати та критично аналізувати інформацію щодо поточного стану та трендів розвитку, інструментів та методів досліджень, наукових та інноваційних проєктів з комп'ютерних наук.

ПРН 11. Організовувати і здійснювати освітній процес у сфері комп'ютерних наук, його наукове, навчально-методичне та нормативне забезпечення, застосовувати ефективні методики викладання навчальних дисциплін.

3. Очікувані результати навчання

В результаті вивчення дисципліни студент повинен

ЗНАТИ:

- фундаментальні поняття та базові визначення технології програмування;
- способи розробки прикладного програмного забезпечення, основи технології структурного підходу до програмування, концепцію і складові частини об'єктно-орієнтованого програмування;

- сучасні технології програмування на об'єктно-орієнтованих високого рівня мовах програмування (Python);
- елементарні поняття, уявлення щодо самостійної розробкою програмного забезпечення для подальшого виконання курсових і випускних кваліфікаційних робіт;
- базові уявлення щодо основних етапів проектування, планування, кодування, налагодження, тестування і експлуатації комп'ютерних програм;
- базові принципи організації колективної роботи по розробці великих програмних проектів.

ВМІТИ:

- виконувати кодування, налагодження і тестування окремих програмних модулів і програмного додатку в цілому;
- розробляти схеми модулів, програм, визначати план розробки програмних модулів;
- кваліфіковано готувати тестові завдання для налагодження окремих програмних модулів і програмного додатку в цілому.
- використовувати засоби колективної розробки програм на мові Python.

4. Засоби діагностики результатів навчання

Засобами оцінювання та методами демонстрування результатів навчання при опануванні дисципліною «Сучасні технології програмування»:

- іспит;
- індивідуальні завдання з практичних робіт;
- презентації результатів виконаних завдань розрахункової роботи.

Під час викладання дисципліни «Сучасні технології програмування» використовуються наступні засоби діагностики.

- Поточний контроль знань під час виконання практичних робіт: усне опитування студентів за основними питаннями, контроль результативності виконання практичних завдань за темою роботи.
- Оцінка презентації та доповіді за результатом виконання розрахункової роботи.
- Іспит.

5. Критерії оцінювання результатів навчання

Загальний принцип оцінювання підсумкових знань студента з курсу «Сучасні технології програмування» полягає в оцінці виконання завдань практичної роботи протягом навчального семестру, оцінці самостійного виконання (під керівництвом викладача) розрахункової роботи та оцінки контрольного заходу у формі екзамену, у результаті котрих студент має сумарну оцінку в балах. Форма проведення іспиту – письмова. Максимальна кількість балів, що може отримати студент за виконання письмової роботи, під час семестрового екзамену – 60 (див. табл.).

Поточний контроль для денної форми навчання									Поточний контроль	Іспит	Максимальний бал
Пр1	Пр2	Пр3	Пр4	Пр5	Пр6	Пр7	Пр8	РР			
4	4	4	4	4	4	5	5	6	40	60	100
2	2	2	3	3	3	3	3	3	24		

Примітки: 1) Пр1, Пр2 і т.д. практичні заняття;
2) У чисельнику максимальний бал – при своєчасному та правильному виконанні, у знаменнику – мінімальний (при правильному, але несвоечасному виконанні)

Оцінювання знань студента при здачі іспиту здійснюється за 100 бальною шкалою.

Оцінка	
За 100-бальною шкалою	Для екзамену, курсового проекту(роботи), практики, диференційованого заліку, кваліфікаційного екзамену, випускної кваліфікаційної (дипломної) роботи (проекту)
90-100	відмінно
74-89	добре
60-73	задовільно
0-59	незадовільно

6. Програма навчальної дисципліни

6.1. Основні теми дисципліни

Тема 1. Введення. Неформальний вступ до мови програмування Python.

Історія створення. Типова структура Python програми: Лексична структура, синтаксична структура. Змінні. Типи даних (скалярні цілі, дійсні). Операції, оператори. Колекції даних: рядки, списки, кортежі, словники, базові операції з ними.

Тема 2. Функції в Python.

Параметри та аргументи функцій. Локальні та глобальні змінні. Рекурсивний виклик функцій.

Тема 3. Класи та об'єкти.

Основні відомості. Опис класу. Області видимості та простір імен. Доступ до елементів класу. Підтримка ООП в Python: інкапсуляція, спадкування, поліморфізм. Атрибути класу, атрибути об'єкту. Закриті атрибути. Визначення та оголошення функцій та класів. Типи та класи в Python.

Тема 4. Організація пакетів (модулів) в Python.

Модулі та пакети. Створення модулів. Імпорт модулів. Пакети. Інсталяція пакетів. Уведення - виведення даних.

Тема 5. Файлові об'єкти в Python.

Об'єкт-файл. Базові властивості. Методи об'єктів-файлів. Операції читання та запису файлів. Помилки та виняткові ситуації. Обробка виняткових ситуацій. Генерування виняткових ситуацій.

Тема 6. Типові пакети Python: математичні пакети.

Математичні пакети *math*, *random*, *statistics*, *numpy*. Масиви – створення та індексація. Операції та функції з масивами. Двовимірні масиви, форма масивів.

Тема 7. Типові пакети Python: графічні пакети.

Пакет *Matplotlib* - графічне відображення даних. Створення зображень: заголовки, легенди, тексти. Графіки, кругові діаграми, гістограми. Відображення даних в 3D.

Тема 8. Типові пакети Python: обробка тексту.

Пакети обробки тексту. Виділення порядків, пошук порядків, заміщення. Розбиття рядків, об'єднання рядків. Основні операції з рядками. Регулярні вирази.

Тема 9. Типові пакети Python: створення графічних інтерфейсів.

Пакети створення інтерфейсів користувача. Пакет *tkinter*: основні елементи (вікна, віджети), опрацювання подій, меню, вбудовування графіків.

Тема 10. Організація процесу проектування програмного забезпечення

Ведення в технологію програмування, програмну інженерію. Поняття програмного засобу. Технологія програмування та основні етапи її розвитку. Проблеми розробки складних програмних систем. Життєвий цикл програмного засобу. Моделі розробки: каскадний, з проміжним контролем, спіральна і т.д .

Тема 11. Технології створення програмного коду

Бібліотеки стандартних компонентів, бібліотеки об'єктів. Проектування інтерфейсу з користувачем. Статичні, полустатичні і динамічні типи даних. Прості і складові типи даних, операція кваліфікації. Технології розподілених обчислень: RPC, RMI, Corba, DCOM. Огляд і класифікація засобів підтримки колективної розробки ПО. Програмні засоби планування і управління процесом розробки. Мережеві графіки та діаграми робочого процесу. Сценарії виконання робіт. Застосування систем управління документами. CASE-технології.

Тема 12. Методи налагодження та тестування програм.

Категорії програмних помилок. Типи тестів. Тестування на етапі планування. Тестування на етапі проектування. Тестування "білого ящика" на стадії кодування. Регресійне тестування. Тестування "чорного ящика". Розробка тестів.

6.2. Теми практичних занять

№	Назва теми	Кількість годин, денна форма
1	Практичне заняття №1. (Тема 1). Вступ. Керуючі конструкції у Python: <i>While, For, If</i> .	4
2	Практичне заняття №2. (Тема 2). Мова Python: базові функції.	4
3	Практичне заняття №3. (Тема 2). Мова Python: колекції даних: рядки, списки, кортежі, словники.	4
4	Практичне заняття №4. (Тема 3, 5). Мова Python, робота із файлами, та дослідження рекурсивних функцій.	4
5	Практичне заняття №5. (Тема 4). Мова Python: модулі, пакети.	4
6	Практичне заняття №6. (Тема 3). Мова Python: класи, спадкування.	4
7	Практичне заняття №7. (Тема 6). Математичний пакет NumPy.	4
8	Практичне заняття №8. (Тема 7). Графічний пакет Matplotlib, візуалізація даних.	4
Усього годин:		32

6.3. Теми лабораторних занять

Не передбачено навчальним планом

6.4. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Тема 1. Введення. Неформальний вступ до мови програмування Python.	10
2	Тема 2. Функції в Python.	10
3	Тема 3. Класи та об'єкти.	10
4	Тема 4. Організація пакетів (модулів) в Python.	10
5	Тема 5. Файлові об'єкти в Python.	10
6	Тема 6. Типові пакети Python: математичні пакети.	10
7	Тема 7. Типові пакети Python: графічні пакети.	10
8	Тема 8. Типові пакети Python: обробка тексту.	10
9	Тема 9. Типові пакети Python: створення графічних інтерфейсів.	10
10	Тема 10. Організація процесу проектування програмного забезпечення.	10
11	Тема 11. Технології створення програмного коду.	8
12	Тема 12. Методи налагодження та тестування програм.	8
Усього годин:		116

6.5. Індивідуальне завдання

Тематика індивідуального завдання за розрахунковою роботою пов'язана із створенням програмного пакету мовою Python для вирішення заданого класу задачі математичного аналізу визначеним чисельним методом.

7. Література

7.1. Основна

1. Програмування числових методів мовою Python: підруч. / А. В. Анісімов, А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий ; за ред. А. В. Анісімова. – К. : Видавничо-поліграфічний центр "Київський університет", 2014. – 640 с.
2. Програмування числових методів мовою Python: навч. посіб. / А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий, Є. В. Глушко ; за ред. А. В. Анісімова. – К. : Видавничо-поліграфічний центр "Київський університет", 2013. – 463 с.
3. Основи програмування Python : Підручник для студ. спеціальності 122 «Комп'ютерні науки» / А.В.Яковенко; КПП.- Київ: КПП, 2018 . – 195 с.
4. Основи інформатики та технологій програмування: навчальний посібник / Рогоза М.Є., Рамазанов С.К., Велігура А.В., Танченко С. М. - Луганськ: Вид-во СНУ ім. В.Даля, 2012. - 568 с.
5. Навчальний посібник з дисципліни «Технології розробки програмного забезпечення» для студентів спеціальності 123 «Комп'ютерна інженерія» /Дегтярьова Л.М., Гроза П.М., Сомов С.В. – Полтава: ПолтНТУ, 2017. – 218 с.

7.2. Допоміжна

1. Lutz M. Learning Python, 5-th ed. – O'Reilly Media, Inc., 5th edition, 2013.- 1593 p. – ISBN 1449355730
2. Python Cookbook: Recipes for Mastering Python 3. – O'Reilly and Associates; 3rd edition, 2013.- 704 p. – ISBN 1449340377

7.3. Методична

1. Методичні вказівки до виконання практичних робіт з дисципліни «Сучасні технології програмування» для студентів денної та заочної форм навчання спеціальності 122 Комп'ютерні науки (плануються до видання)

8. Інформаційні ресурси

1. Життєвий цикл програмного забезпечення. <https://uk.wikipedia.org/wiki>
2. SWEBOK. <https://ru.wikipedia.org/wiki/SWEBOK>